Expert Systems with Applications 38 (2011) 10524–10531

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Weighted SCL model for adaptation of sentiment classification

Songbo Tan [a,*], Yuefen Wang [b]

[a] Key Laboratory of Network, Institute of Computing Technology, Chinese Academy of Sciences, China
[b] Information Center, Chinese Academy of Geological Sciences, China

ARTICLE INFO

Keywords:
Sentiment analysis
Opinion mining
Information retrieval
Data mining

ABSTRACT

In recent years, Structural Correspondence Learning (SCL) is regarded as one of the most promising techniques for transfer learning. The main idea behind SCL model is to identify correspondences among features from different domains by modeling their correlations with pivot features. However, SCL model treats each feature as well as each instance by an equivalent-weight strategy. From the perspective of feature, this strategy fails to overcome the adverse influence of high-frequency domain-specific (HFDS) features: they occupy a relative large portion of *weight* in classification model, while hardly carry corresponding sentiment information. From the other perspective, the equivalent-weight strategy of SCL model does not take into account the labels ("positive" or "negative") of labeled instance and the labels of pivot features: positive pivot features tend to occur more frequently in positive instances and vice versa. To address the two issues effectively, we proposed a weighted SCL model (W-SCL), which weights the features as well as the instances. More specifically, W-SCL assigns a smaller weight to HFDS features and assigns a larger weight to instances with the same label as the involved pivot feature. The experimental results indicate that proposed W-SCL model could overcome the adverse influence of HFDS features, and leverage knowledge from labels of instances and pivot features.

1. Introduction

In the community of sentiment analysis (Tang, Tan, & Cheng, 2009), we are often confronted with sentiment classification tasks where we have a sufficient amount of labeled data in one old (or source) domain, but have no labeled data in another new (or target) domain that we are interested in. Accordingly, the only feasible way is to train a classifier using the source-domain data and transfer it to the target domain. This is so-called "transfer learning" problem.

However, transferring a sentiment classifier from one source domain to another target domain is still far from a trivial work, because sentiment expression often behaves with strong domain-specific nature. In another word, in each different domain, the sentiments are apt to be expressed by its own domain-specific features. For example, "rise" and "rebound" are often used to express positive sentiment for stock review; while "luxury" and "classical" are often employed to convey positive sentiment for house review.

Up to this time, many researchers have proposed techniques to address this problem, such as classifiers adaptation, generalizable features detection and so on (Ando & Zhang, 2005; Blitzer, McDonald, & Pereira, 2006, 2007; Chelba & Acero, 2004; Dai, Xue, Yang, & Yu, 2007; Daume & Marcu, 2006; Du, Tan, Cheng, & Yun, 2010; Jiang & Zhai, 2007; Li & Bilmes, 2007; Raina, Ng, & Koller, 2005; Tan, Wu, Tang, & Cheng, 2007; Tan, Cheng, Wang, & Xu, 2009; Wu et al., 2009). Among these techniques, SCL (Structural Correspondence Learning) (Ando & Zhang, 2005; Blitzer et al., 2006, Blitzer, Dredze, & Pereir, 2007) is regarded as a promising method to tackle transfer-learning problem. The main idea behind SCL model is to identify correspondences among features from different domains by modeling their correlations with pivot features (or generalizable features). Pivot features are features which behave in the same way for discriminative learning in both domains. Non-pivot features from different domains which are correlated with many of the same pivot features are assumed to correspond, and we treat them similarly in a discriminative learner.

For example, the word "good" or "excellent" occurs frequently in stock review as well as in house review, and so they can be regarded as pivot features. In the stock review, both "rise" and "rebound" have high correlation with "good" or "excellent"; in the house review, both "luxury" and "classical" have high correlation with "good" or "excellent". In this case, we think there exists a correspondence between "rise" (or "rebound") and "luxury" (or "classical"). After learning a classifier for stock reviews, when we see a house feature like "luxury" or "classical", we know it should behave in a roughly similar manner to "rise" or "rebound".

However, SCL model treats each feature as well as each instance by an equivalent-weight strategy. From the perspective of feature,

* Corresponding author. Address: Key Laboratory of Network, P.O. Box 2704, Beijing 100190, China. Tel.: +86 1062600928; fax: +86 1062600905.
E-mail address: tansongbo@software.ict.ac.cn (S. Tan).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.02.106

this strategy fails to overcome the adverse influence of high-frequency domain-specific (HFDS) features. For example, the words "stock" or "market" occurs frequently in most of stock reviews, so these non-sentiment features tend to have a strong correspondence with pivot features. As a result, the representative ability of the other sentiment features will inevitably be weakened to some degree.

In this paper, we present an empirical study as well as a formal analysis on adverse influences of HFDS features. The analysis indicates that very few HFDS features occupy a relative large portion of *weight* in classification model, while hardly carry corresponding sentiment information. To address this issue, we proposed Frequently Exclusively-occurring Entropy (FEE) to pick out HFDS features, and proposed a feature-weighted SCL model (FW-SCL) to adjust the influence of HFDS features in building correspondence. The main idea of FW-SCL is to assign a smaller weight to HFDS features so that the adverse influence of HFDS features can be decreased.

From the other perspective, the equivalent-weight strategy of SCL model ignores the labels ("positive" or "negative") of labeled instance. Obviously, this is not a good idea. In fact, positive pivot features tend to occur in positive instances, so the correlations built on positive instances are more reliable than the correlations built on negative instances; and vice versa. Consequently, utilization of labels of instance and pivot features can decrease the adverse influence of some co-occurrences, such as co-occurrences involved with positive pivot features and negative instances, or co-occurrences involved with negative pivot features and positive instances.

In order to take into account the labels of labeled instance, we proposed an instance-weighted SCL model (IW-SCL), which assigns a larger weight to instances with the same label as the involved pivot feature. In this time, we obtain a combined model: feature-weighted and instance-weighted SCL model (FWIW-SCL). For the sake of convenience, we simplify "FWIW-SCL" as "W-SCL" in the rest of this paper.

To investigate the effectiveness of proposed approach, we conducted an extensive experiment on three Chinese domain-specific tasks, including education reviews, stock reviews, and computer reviews. The experimental results indicate that proposed W-SCL model could overcome the adverse influence of HFDS features, and leverage knowledge from labels of instances and pivot features.

The rest of this paper is constructed as follows: Next section presents related work. Traditional SCL model is provided in Section 3. Proposed method is described in Sections 4 and 5. Experimental results are given in Section 6. Finally Section 7 concludes this paper.

2. Related work

In this section, we present the related work about Sentiment Classification (Aue & and Gamon, 2005; Chaovalit & and Zhou, 2005; Mullen & Collier, 2004; Pang, Lee, & Vaithyanathan, 2002; Tan & Zhang, 2008; Tang et al., 2009), Semi-supervised Learning (Joachims, 1999; Lanquillon, 2000; Nigam, McCallum, Thrun, & Mitchell, 1998) and Transfer Learning (Ando & Zhang (2005), Blitzer et al. (2006, 2007), Chelba & Acero (2004), Dai et al. (2007), Daume & Marcu (2006), Du et al. (2010), Jiang & Zhai (2007), Li & Bilmes (2007), Raina et al. (2005), Tan et al. (2007), Tan et al. (2009), Wu et al. (2009)).

In most cases, the use of statistical or machine learning techniques has proven to be successful for sentiment classification, such as Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machines (SVM) (Chaovalit & and Zhou, 2005; Mullen & Collier, 2004; Pang et al., 2002).

Pang et al. (2002) conducted an extensive experiment on movie reviews using three traditional supervised machine-learning methods (i.e., Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machines (SVM)). Her results indicate that standard machine learning techniques definitively outperform human-produced baselines.

Mullen and Collier (2004) employed Support Vector Machines (SVM) to bring together diverse sources of potentially pertinent information, including several favorability measures for phrases and adjectives and, where available, knowledge of the topic of the text. Models using the features introduced are further combined with uni-gram models and lemmatized versions of the unigram models.

Another related work is involved with semi-supervised learning, which trains a classification model using both labeled and unlabeled data. In recent years, semi-supervised learning has received considerable attention in the community of text classification (Joachims, 1999; Lanquillon, 2000; Nigam et al., 1998).

Nigam et al. (1998) introduced an EM-like approach that combines Expectation Maximization (EM) algorithm with Naive Bayes classifier. The result of combining these two is an algorithm that extends conventional text learning algorithms by using EM to dynamically derive pseudo labels for unlabeled documents during learning thereby providing a way to incorporate unlabeled data into supervised learning.

Following this direction, Lanquillon (2000)described a general framework for extending any text-learning algorithm to utilize unlabeled documents in addition to labeled document using an Expectation–Maximization-like scheme. Aue and and Gamon (2005) employed this method to address sentiment classification.

Joachims (1999) modified SVM to exploit the unlabeled data (often called TSVM). TSVM expects to find a low-density area of data and constructs a linear separator in this area so that the margin over both the labeled data and the unlabeled data can be maximized.

Recently, transfer learning has been recognized as an important topic in machine learning research. Several researchers have proposed new approaches to solve the problems of transfer learning (Ando & Zhang (2005), Blitzer et al. (2006, 2007), Chelba & Acero (2004), Dai et al. (2007), Daume & Marcu (2006), Du et al. (2010), Jiang & Zhai (2007), Li & Bilmes (2007), Raina et al. (2005), Tan et al. (2007), Tan et al. (2009), Wu et al. (2009)). However, up to this time, only a little work has been conducted on sentiment transfer learning.

Daume and Marcu (2006) studied the domain-transfer problem in statistical natural language processing. They consider the common case in which labeled out-of-domain data is plentiful, but labeled in-domain data is scarce. Then they introduce a statistical formulation of this problem in terms of a simple mixture model and present an instantiation of this framework to Maximum Entropy (ME) and their linear chain counterparts.

Blitzer et al. (2007) attempted to attack sentiment domain-transfer problem using structural correspondence learning (SCL) (Ando & Zhang, 2005; Blitzer et al., 2006). He suggested selecting pivots based not only on their common frequency but also according to their mutual information with the source labels. However, his method does not consider the adverse influence of high-frequency domain-specific (HFDS) features.

Dai et al. (2007) proposed an EM-based Naive Bayes classifier for domain-transfer problem. In order to transfer the model from the source domain to the target domain, he used KL-divergence to decide the trade-off parameters between the source-domain data and the target-domain data. In fact, KL-divergence only serves as a constant parameter that impacts a much larger weight on the

target-domain data when estimating the class probability and word probability.

3. SCL model

Structural Correspondence Learning (SCL) attempts to learn a correspondence relationship between one source domain and another target domain. Its work situation is that both domains have enough unlabeled data, but only the source domain has labeled data. Generally speaking, SCL can be divided into four steps: choosing pivot features, learning pivot predictors, computing principal pivot features and training a classifier on augmented feature space. In the following, we detail these steps.

First we need to pick out pivot features. Pivot features occur frequently in both the source and the target domain. But not all of this kind of words can serve as pivot features for sentiment transfer, such as stop-words: "the", "a", "this", and "that". In the community of sentiment analysis, generalizable sentiment words are good candidates for pivot features, such as "good" and "excellent". In the rest of this paper, we use K to stand for the number of pivot features.

Second, we need to compute the pivot predictors (or mapping vectors) using selected pivot features. The pivot predictors are the key job, because they directly decide the performance of SCL. For each pivot feature k, we use a loss function L_k like SVMs,

$$L_k = \sum_i (p_k(x_i) w^T x_i - 1) + \lambda \|w\|, \tag{1}$$

$$p_k(x_i) = \begin{cases} 1 & \text{if } x_{ik} > 0, \\ -1 & \text{otherwise}, \end{cases}$$

where x_i is an example from the source domain or the target domain, and the weight vector w encodes the covariance of the non-pivot features with the pivot feature k. If the weight given to the z'th feature by the k'th pivot predictor is positive, then z is positively correlated with the pivot feature k. If two non-pivot features are correlated in the same way with many of the same pivot features, then they have a high degree of correspondence.

The third step is to calculate the principal pivot predictors of the original pivot features. From the perspective of statistics, the principal pivot predictors can capture the variance of the original pivot predictor space as best as possible in $K'(<K)$ dimensions. Given W as the matrix whose columns are the pivot predictor vectors. Then let $W = UDV^T$ as the singular value decomposition of W, so that $\theta = U^T_{[1:K',:]}$ is the matrix whose rows are the top left singular vectors of W.

Finally we use the augmented space $[x^T, x^T\theta]^T$ to train the classifier on the source labeled data and predict the examples on the target domain. If we have learned the pivot vectors well, then θ should encode correspondences among features from different domains which are important for the training of sentiment classifier, and the classifier we train using these new features on the source domain will perform well on the target domain. The outline of SCL algorithm is displayed in Fig. 1.

1 Load the source-domain data (D^o), the target-domain unlabeled data (D^n), and parameters, λ;
2 Pick out K pivot features;
3 For each pivot pvt_k,
3.1 Calculate its mapping vector w_k using formula (1)
4 Compute the $K'(K'<K)$ principal pivot vectors using SVD decomposition.
5 Train a new classifier using the augmented space $[x^T, x^T\theta]^T$.

Fig. 1. The outline of SCL algorithm.

4. Feature-weighted SCL model

In this section, we first present an empirical study as well as a formal analysis of adverse influences of HFDS features. Then we proposed Frequently Exclusively-occurring Entropy (FEE) to pick out HFDS features. Lastly, we proposed feature-weighted SCL model (FW-SCL) to adjust the influence of HFDS features in building correspondence.

4.1. Empirical study on HFDS features

In this section, we take the negative class of stock review (Sto) as an example, and have a close look at how prevalence HFDS features behave in their own domain. In order to describe the weight of each word in classification model, we calculate its normalized prototype vector υ for the negative class using the following formula,

$$\upsilon = \frac{\sum_i d_i}{\|\sum_i d_i\|_2}, \tag{2}$$

$$d_{i,j} = \frac{d^t_{i,j} \times \log(N/n_t)}{(\sum_{j \in d_i} (d^t_{i,j} \times \log(N/n_t))^2)^{1/2}}, \tag{3}$$

where N is the total number of training documents, n_j is the number of documents containing the word j, and $d^t_{i,j}$ indicates the occurrences of word j in document i. Before discussion, we give some definitions, i.e., weight ratio ($R_{Z,\upsilon}$), word ratio ($R_{Z,w}$), and sentiment weight ratio ($R_{S,Z,\upsilon}$),

$$R_{Z,\upsilon} = \frac{\sum_{i \in Z} \upsilon_i}{\sum_i \upsilon_i}, \tag{4}$$

$$R_{Z,w} = \frac{|Z|}{|Z^*|}, \tag{5}$$

$$R_{S,Z,\upsilon} = \frac{\sum_{i \in (S^* \cap Z)} \upsilon_i}{\sum_{i \in S^*} \upsilon_i}, \tag{6}$$

where Z indicates a word set drawn from the dictionary Z^* of the given domain, and S^* indicates the sentiment dictionary of the given domain.

From Fig. 2, we can observe that a few top-frequency words occupy a large portion of *weight* in classification model while carry very few sentiment information. For example, the words with *weight* bigger than 0.06 amount to 1% of the dictionary Z^* in stock review, occupy 25% of *weight* in classification model while only

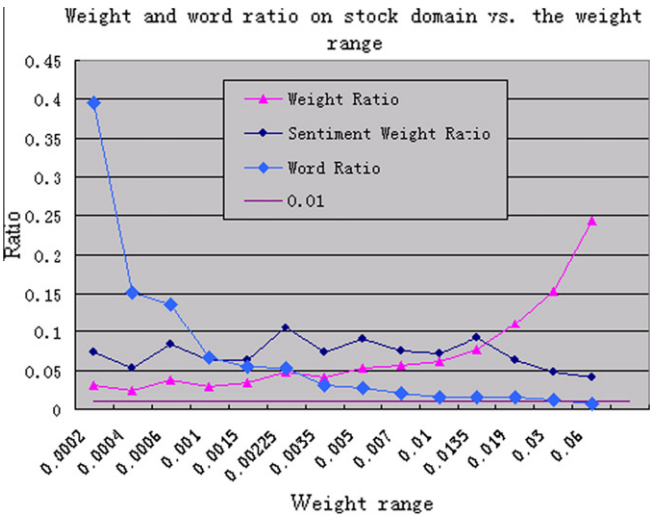

Fig. 2. Weight and word ratio on stock domain vs. the weight range.

carry less than 5% of sentiment *weight* in classification model. That is to say, one percent of the dictionary Z^* occupy more than 25% of *weight* in classification model while only carry less than 5% of sentiment *weight* in classification model. This trend is even more pronounced for words with *weight* bigger than 0.03, which only sum to 2% of the dictionary Z^* in stock review, occupy nearly 40% of *weight* in classification model, while only carry less than 10% of sentiment *weight* in classification model.

As a result, we may make a conclusion that a small portion of top-frequency words occupy a relative large portion of *weight* in classification model, but hardly carry corresponding sentiment information. In another word, very few top-frequency words degrade the representative ability of classification model for sentiment classification.

4.2. Formal analysis of adverse influences of HFDS features

According to formula (1), we can compute a correspondence vector $\boldsymbol{w}_k$ for each pivot feature pvt_k. Like SVMs, $\boldsymbol{w}_k$ can be formulated as following,

$$w_k = \sum_i p_k(x_i)\alpha_i x_i = \sum_{p_k(x_i)>0} \alpha_i x_i - \sum_{p_k(x_i)<0} \alpha_i x_i, \tag{7}$$

where α_i corresponds to supporting vectors (document vectors).

For the sake of convenience, we use to two prototype vectors, i.e., υ^+ and υ^- to stand for the first and the second item respectively,

$$\upsilon^+ = \sum_{p_k(x_i)>0} \alpha_i x_i,$$
$$\upsilon^- = \sum_{p_k(x_i)<0} \alpha_i x_i.$$

We assume the feature h is the only HFDS feature in the vectors υ^+ and υ^-. So we can write down the υ^+ and υ^- as following,

$$\upsilon^+ = \left(\upsilon_1^+, \upsilon_2^+, \ldots, \upsilon_h^+, \ldots, \upsilon_N^+\right),$$
$$\upsilon^- = \left(\upsilon_1^-, \upsilon_2^-, \ldots, \upsilon_h^-, \ldots, \upsilon_N^-\right),$$

where $c_h^+ = c_h^- + \varepsilon$. And given a document vector $\boldsymbol{d}_i$, then we can compute its score by inner product,

$$\begin{aligned} s &= d_i\upsilon^+ - d_i\upsilon^- = \sum_j d_{ij}\upsilon_j^+ - \sum_j d_{ij}\upsilon_j^- \\ &= \sum_{j\neq h} d_{ij}\upsilon_j^+ + d_{ih}\upsilon_h^+ - \sum_{j\neq h} d_{ij}\upsilon_j^- - d_{ih}\upsilon_h^- \\ &= \sum_{j\neq h} d_{ij}\upsilon_j^+ - \sum_{j\neq h} d_{ij}\upsilon_j^- + d_{ih}\upsilon_h^+ - d_{ih}\upsilon_h^- \\ &= \sum_{j\neq h} d_{ij}\upsilon_j^+ - \sum_{j\neq h} d_{ij}\upsilon_j^- + d_{ih}\left(\upsilon_h^+ - \upsilon_h^-\right) = \varphi + d_{ih}\left(\upsilon_h^+ - \upsilon_h^-\right) \\ &= \varphi + d_{ih}\varepsilon, \end{aligned} \tag{8}$$

where

$$\varphi = \sum_{j\neq h} d_{ij}\upsilon_j^+ - \sum_{j\neq h} d_{ij}\upsilon_j^-.$$

According to above formula, we can analysis two condition sets where HFDS features will degrade the performance of SCL algorithm:

(1) $\boldsymbol{d}_{ik} > \boldsymbol{0}$, $\boldsymbol{d}_{ih} > \boldsymbol{0}$, $\boldsymbol{\phi} > \boldsymbol{0}$ **and** $\boldsymbol{\varepsilon} < \boldsymbol{0}$
The conditions "$\boldsymbol{d}_{ik} > \boldsymbol{0}$" and "$\boldsymbol{d}_{ih} > \boldsymbol{0}$" indicate both the pivot feature pvt_k and the HFDS feature h occur in document $\boldsymbol{d}_i$. In this condition set, if $|\boldsymbol{\phi}| > |\boldsymbol{d}_{ih}\boldsymbol{\varepsilon}|$, document $\boldsymbol{d}_i$ will be correctly classified while if $|\boldsymbol{\phi}| \leqslant |d_{ih}\boldsymbol{\varepsilon}|$, document $\boldsymbol{d}_i$ will be misclassified. Because the feature h is a HFDS feature, the event "$|\boldsymbol{\phi}| \leqslant |\boldsymbol{d}_{ih}\boldsymbol{\varepsilon}|$" absolutely will not be a small-probability event. As a result, there exists a quite large probability that HFDS features may degrade the performance SCL algorithm.

(2) $\boldsymbol{d}_{ik} \leqslant \boldsymbol{0}$, $\boldsymbol{d}_{ih} > \boldsymbol{0}$, $\boldsymbol{\phi} \leqslant \boldsymbol{0}$ **and** $\boldsymbol{\varepsilon} > \boldsymbol{0}$
The condition "$\boldsymbol{d}_{ik} \leqslant \boldsymbol{0}$" indicates that the pivot feature pvt_k does not occur in document $\boldsymbol{d}_i$, and "$\boldsymbol{d}_{ih} > \boldsymbol{0}$" indicates that the HFDS feature h occurs in document $\boldsymbol{d}_i$. In this condition set, if $|\boldsymbol{\phi}| \geqslant |\boldsymbol{d}_{ih}\boldsymbol{\varepsilon}|$, document $\boldsymbol{d}_i$ will be correctly classified while if $\boldsymbol{\phi} < |\boldsymbol{d}_{ih}\boldsymbol{\varepsilon}|$, document $\boldsymbol{d}_i$ will be misclassified. Like analysis in condition set (1), there also exists a quite large probability that HFDS features may degrade the performance SCL algorithm.

More generally, there is more than one HFDS feature in one domain dataset and so the HFDS features will occupy a large portion of weight in each document vector as well as in vectors υ^+ and υ^-. Consequently, although most of the HFDS features often contain very little sentiment information, they tend to play a great adverse role in the sentiment classification.

4.3. Measure to pick out HFDS features

In order to pick out HFDS features, we proposed Frequently Exclusively-occurring Entropy (FEE). Our measure includes two criteria: occur in one domain as frequently as possible, while occur on another domain as rarely as possible. To satisfy this requirement, we proposed the following formula:

$$\begin{aligned} f_w &= \log\left(\max(P_o(w), P_n(w)) \times \frac{\max(P_o(w), P_n(w))}{\min(P_o(w), P_n(w))}\right) \\ &= \log(\max(P_o(w), P_n(w))) + \log\left(\frac{\max(P_o(w), P_n(w))}{\min(P_o(w), P_n(w))}\right), \end{aligned} \tag{9}$$

where $P_o(w)$ and $P_n(w)$ indicate the probability of word w in the source domain and the target domain respectively:

$$P_o(w) = \frac{(N_o(w) + \alpha)}{(N_o + 2\cdot\alpha)}, \tag{10}$$

$$P_n(w) = \frac{(N_n(w) + \alpha)}{(N_n + 2\cdot\alpha)}, \tag{11}$$

where $N_o(w)$ and $N_n(w)$ is the number of examples with word w in the source domain and the target domain respectively; N_o and N_n is the number of examples in the source domain and the target domain respectively. In order to overcome overflow, we set $\alpha = 0.0001$ in our experiment reported in Section 6.

In the extreme case when $\min(P_o(w), P_n(w)) = 0$, above formula can not work. As a result, we modify our formula as following,

$$f_w = \log(\max(P_o(w), P_n(w))) + \log\left(\frac{\max(P_o(w), P_n(w))}{\min(P_o(w), P_n(w)) + \beta}\right), \tag{12}$$

where β is set as 1.0 in our experiment.

To better understand this measure, let's take a simple example (see Table 1). Given a source dataset with 1000 documents and a target dataset with 1000 documents, 12 candidate features, and a task to pick out 2 HFDS features. According to our understanding, the best choice is to pick out w_4, and w_8.

According to formula (12), fortunately, we successfully pick out w_4, and w_8. This simple example validates the effectiveness of proposed FEE formula.

4.4. Feature-weighted SCL model

To adjust the influence of HFDS features in building correspondence, we proposed feature-weighted SCL model (FW-SCL),

$$L_k = \sum_i \left(p_k(x_i)\sum_l \delta_l w_l x_{il} - 1\right) + \lambda\|w\|, \tag{13}$$

Table 1
A simple example for FEE.

No.	$N_o(w)$	$N_n(w)$	FEE	
			Score	Rank
1	**100**	**100**	−2.3025	6
2	**100**	**90**	−2.1971	4
3	100	45	−1.5040	3
4	100	4	**0.9163**	1
5	**50**	**50**	−2.9956	8
6	**50**	**45**	−2.8903	7
7	50	23	−2.2192	5
8	50	2	**0.2231**	2
9	4	4	−5.5214	11
10	4	3	−5.2337	10
11	4	2	−4.8283	9
12	1	1	−6.9077	12

$$p_k(x_i) = \begin{cases} 1 & if\ x_{ik} > 0, \\ -1 & otherwise, \end{cases}$$

$$\delta_l = \begin{cases} \eta & if\ l \in Z_{HFDS}, \\ 1 & otherwise, \end{cases}$$

where Z_{HFDS} indicates the HFDS feature set and η ($\eta \in [0,1]$) is the parameter to control the weight of HFDS features. When "$\eta = 0$", it indicates that no HFDS features are used to build the correspondence vectors; while "$\eta = 1$" indicates that all features are equally used to build the correspondence vectors, that is to say, proposed FW-SCL algorithm is simplified as traditional SCL algorithm. Consequently, proposed FW-SCL algorithm could be regarded as a generalized version of traditional SCL algorithm.

5. Instance-weighted SCL model

The traditional SCL model does not take into account the labels ("positive" or "negative") of instances on the source domain and pivot features. Although the labels of pivot features are not given at first, it is very easy to obtain these labels because the number of pivot features is typically very small.

Obviously, positive pivot features tend to occur in positive instances, so the correlations built on positive instances are more reliable than the correlations built on negative instances; and vice versa. But we cannot think the correlations (involved with positive pivot) built on negative instances are completely useless. There are two reasons accounting for this viewpoint: the first is that we cannot guarantee that the labels of source-domain instances are absolutely correct; the second is that even on negative (positive) instances, there are some certain positive (negative) opinions.

As a result, the ideal choice is to assign a larger weight to the instances with the same label with the involved pivot feature, while assign a smaller weight to the instances with the different label with the involved pivot feature. This strategy can make correlations more reliable. This is the key idea of instance-weighted SCL model (IW-SCL). We can write down the IW-SCL model as following,

$$L_k = \gamma \cdot \sum sign(lbl(pvt_k), lbl(x_i))(p_k(x_i) w x_i - 1) + (1-\gamma) \cdot \sum (1 - sign(lbl(pvt_k), lbl(x_j)))(p_k(x_j) w x_j - 1) + \lambda \|w\|, \quad (14)$$

$$p_k(x_i) = \begin{cases} 1 & if\ x_{ik} > 0, \\ -1 & otherwise, \end{cases}$$

$$sign(z,y) = \begin{cases} 1 & if\ z = y\ and\ z \neq 0, \\ 0 & otherwise, \end{cases}$$

$$lbl(z) = \begin{cases} 1 & if\ z\ has\ a\ positive\ label, \\ 0 & unknown, \\ -1 & if\ z\ has\ a\ negative\ label, \end{cases}$$

where γ is the parameter that controls the relative weight between the instances with the same labels as the pivot pvt_k and the other instances.

Combining the idea of feature-weighted SCL model, we obtain the feature-weighted and instance-weighted SCL model (FWIW-SCL),

$$L_k = \gamma \cdot \sum sign(lbl(pvt_k), lbl(x_i))(p_k(x_i) \sum_l \delta_l w_l x_{il} - 1) + (1-\gamma) \cdot \sum (1 - sign(lbl(pvt_k), lbl(x_j)))(p_k(x_j) \sum_l \delta_l w_l x_{jl} - 1) + \lambda \|w\|, \quad (15)$$

$$p_k(x_i) = \begin{cases} 1 & if\ x_{ik} > 0, \\ -1 & otherwise, \end{cases}$$

$$\delta_l = \begin{cases} \eta & if\ l \in Z_{HFDS}, \\ 1 & otherwise, \end{cases}$$

$$sign(z,y) = \begin{cases} 1 & if\ z = y\ and\ z \neq 0, \\ 0 & otherwise, \end{cases}$$

$$lbl(z) = \begin{cases} 1 & if\ z\ has\ a\ positive\ label, \\ 0 & unknown, \\ -1 & if\ z\ has\ a\ negative\ label. \end{cases}$$

For the sake of convenience, we simplify "FWIW-SCL" as "W-SCL". The detailed algorithm is outlined in Fig. 3.

Above weighted SCL model only considers the labels of the source-domain instances, and so our next question is whether we can use the "pseudo" labels of the target-domain instances. Although acquisition of accurate labels of the target-domain instances is unfeasible at hand, but it is a very easy job to acquire the "pseudo" labels of a small portion of the target-domain instances. According to the work of Tan et al. (2007), we can use Relative Similarity Ranking (RSR) to label some high-confidence examples from the target-domain instances,

$$\rho_i^{RP} = \frac{\upsilon^P \cdot x_i}{(\upsilon^N \cdot x_i + \upsilon^P \cdot x_i)/2}, \quad (16)$$

$$\rho_i^{RN} = \frac{\upsilon^N \cdot x_i}{(\upsilon^N \cdot x_i + \upsilon^P \cdot x_i)/2}, \quad (17)$$

where ρ^{RP} and ρ^{RN} are the relative positive similarity and the relative negative similarity respectively; υ^P and υ^N are the positive prototype and the negative prototype respectively. According to the evaluation in Tan et al. (2007), we use RSR to label 30% of the unlabeled target-domain examples in our experiment.

1 Load the source-domain data (D^o), the target-domain unlabeled data (D^n), and parameters, λ;
2 Pick out K pivot features;
3 Pick out Z_{HFDS} high-frequency domain-specific features;
4 Label some high-confidence ones from the target domain;
5 For each pivot pvt_k,
  5.1 Calculate its mapping vector w_k using formula (15)
6 Compute the K'(K'<K) principal pivot vectors using SVD decomposition.
7 Train a new classifier using the augmented space.

Fig. 3. The outline of W-SCL algorithm.

6. Experimental results

6.1. Datasets

To validate the effectiveness and robustness of proposed method, we collected three Chinese domain-specific datasets: Education Reviews (Edu, from http://blog.sohu.com/learning/), Stock Reviews (Sto, from http://blog.sohu.com/stock/) and Computer Reviews (Comp, from http://detail.zol.com.cn/). All of these datasets are annotated by three linguists. We use Chinese text POS tool ICTCLAS (http://www.ictclas.org/) to parse and tag these Chinese reviews.

Education Reviews There are 1,012 negative reviews and 254 positive reviews in this corpus. Much larger than Computer Reviews, the average size of reviews is about 600 words, and the cardinality of vocabulary is 19,150.

Stock Reviews This collection consists of 683 negative reviews and 364 positive reviews. Larger than Computer Reviews and smaller than Education Reviews, the average length of reviews is about 460 terms and the different terms amount to 12,674.

Computer Reviews This dataset contains 390 negative reviews and 544 positive reviews about computer. The average length of reviews is about 120 words. This dataset comprises a very small vocabulary-only 4,725 different words.

6.2. Comparison methods

In our experiments, we run two supervised baselines, i.e., Naïve Bayes (NB) (McCallum & Nigam, 1998) and prototype classifier (Tan, Cheng, Ghanem, Wang, & Xu, 2005), which only use one source-domain labeled data as training data.

With regards to Transductive learning baseline, we execute Transductive SVM (TSVM) (Joachims, 1999), which is included in Joachims's SVM-light package. (http://svmlight.joachims.org/). We use a linear kernel and leave all parameters as default. In our experiments, TSVM employs the source-domain labeled data as well as the target-domain unlabeled data.

For transfer-leaning baseline, we implement traditional SCL model (T-SCL) (Ando & Zhang, 2005; Blitzer et al., 2006, 2007). Like TSVM, it makes use of the source-domain labeled data as well as the target-domain unlabeled data.

6.3. Does proposed method work?

To evaluate a sentiment classification system, we use Micro and Macro F1 measure (Tan et al., 2005), which emphasize the performance of the system on common and rare categories respectively.

Table 2
(A) MicroF1 of different methods. (B) MacroF1 of different methods.

	NB	Prototype	TSVM	T-SCL	FW-SCL	W-SCL
Edu- > Sto	0.6704	0.6953	0.7688	0.7965	0.7917	0.8108
Edu- > Comp	0.5085	0.7066	0.6381	0.8019	0.8993	0.9025
Sto- > Edu	0.6824	0.8717	0.7968	0.7712	0.9072	0.9368
Sto- > Comp	0.5053	0.7462	0.6488	0.8126	0.8126	0.8693
Comp- > Sto	0.6580	0.6542	0.7287	0.6523	0.7010	0.7717
Comp- > Edu	0.6114	0.6449	0.7968	0.5976	0.9112	0.9408
Average	0.6060	0.7198	0.7297	0.7387	0.8372	0.8720
Edu- > Sto	0.4553	0.5387	0.7179	0.7621	0.7330	0.7731
Edu- > Comp	0.4696	0.7044	0.6247	0.7730	0.8971	0.9002
Sto- > Edu	0.5867	0.8608	0.7594	0.7712	0.9021	0.9344
Sto- > Comp	0.5025	0.7294	0.6376	0.7915	0.8078	0.8628
Comp- > Sto	0.4148	0.4007	0.6613	0.3947	0.5836	0.7313
Comp- > Edu	0.4105	0.4907	0.7585	0.3740	0.9047	0.9384
Average	0.4732	0.6208	0.6932	0.6444	0.8047	0.8567

To conduct our experiments, we use source-domain data as labeled training set, and use target-domain data as unlabeled set and testing set.

Note that we use 30 pivot features for T-SCL, FW-SCL and W-SCL in the following experiments.

Table 2 shows the results of experiments comparing proposed method with supervised learning, transductive learning and T-SCL. For FW-SCL, the Z_{HFDS} is set to 200 and η is set to 0.1; For W-SCL, the Z_{HFDS} is set to 200, η is set to 0.1, and γ is set to 0.9.

As expected, proposed method FW-SCL does indeed provide much better performance than supervised baselines, TSVM and T-SCL model. For example, the averaged MicroF1 of FW-SCL beats supervised baselines by about 12 percents, beats TSVM by about 11 percents and beats T-SCL by about 10 percents. This trend is even more pronounced for averaged MacroF1. This result indicates that proposed FW-SCL model could overcome the shortcomings of HFDS features in building correspondence vectors.

More surprisingly, instance-weighting strategy can boost the performance of FW-SCL by about 4 percents. This result indicates that the labels of instances and pivot features are very useful in building the correlation vectors. This result also verifies our analysis in Section 5: positive pivot features tend to occur in positive instances, so the correlations built on positive instances are more reliable than the correlations built on negative instances, and vice versa.

Accordingly, we can say that proposed W-SCL model offers a better choice for sentiment-analysis applications that require high-precision classification but hardly have any labeled training data.

From Table 2, we can observe that supervised baselines provide the worst performance. This result indicates that traditional supervised classifiers are not appropriate for transfer learning, because the target-domain data does not comply with the distribution of the source-domain data.

Leveraging knowledge from both the source domain and the target domain, TSVM performs a little better than supervised baselines, only by one percent. The success is attributed to its objective, that is, trying to find a low-density area of data and constructs a linear separator in this area so that the margin over both the labeled data and the unlabeled data can be maximized.

On other hand, however, TSVM is still outperformed by proposed method with wide margin. Intuitively, this is true because TSVM is not a classifier designed for transfer learning.

Although SCL is a method designed for transfer learning, but it cannot provide better performance than TSVM. This result verifies the analysis in Section 4: a small amount of HFDS features occupy a large amount of *weight* in classification model, but hardly carry corresponding sentiment. In another word, very few top-frequency words degrade the representative ability of SCL model for sentiment classification.

6.4. How many HFDS features should we need to pick out?

HFDS features affect the mapping ability of SCL model: they occupy a large amount of *weight* in classification model while hardly carry corresponding sentiment information.

So our question is how many HFDS features we should need to pick out. We vary the number of HFDS features from 0 to 500, run the proposed method W-SCL on six transfer tasks, and finally draw two curves of the averaged performance. Note that for W-SCL, the η is set to 0.1, and γ is set to 0.9.

As can be observed from Fig. 4, with the increase of the number of HFDS features, the performance first grows and then descends. Between 150 and 250, proposed method shows a robust and excellent performance. This observation validates our intuitive

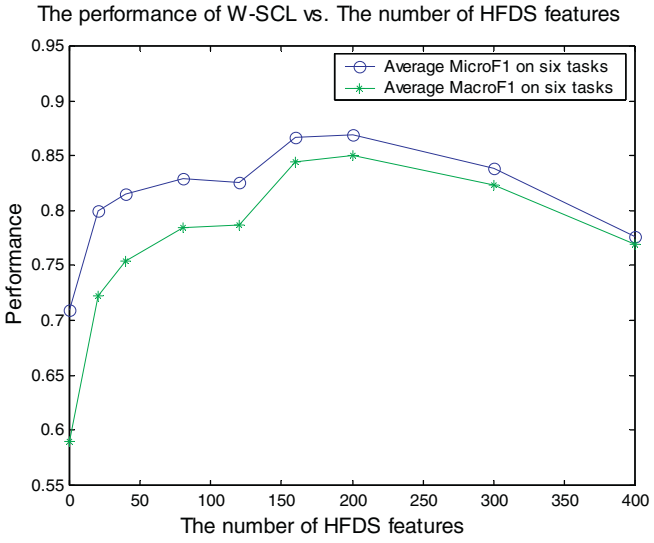

Fig. 4. The performance curves of W-SCL vs. the number of HFDS features.

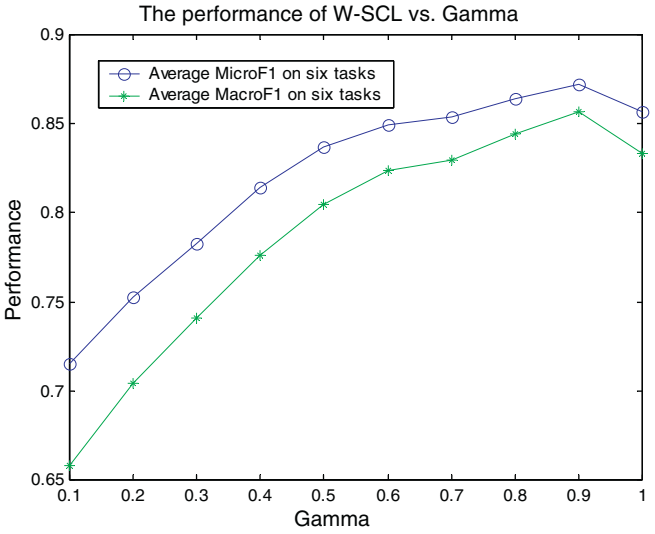

Fig. 6. The performance curves of W-SCL vs. γ.

estimation. Obviously, the number of HFDS features is limited. Hence if the predefined number is bigger than the latent number of HFDS features, some low-frequency features are picked out so that the mapping quality of SCL model is inevitably degraded.

On the contrary, if the predefined number is smaller than the latent number of HFDS features. That is to say, we only pick out a part of HFDS features, so the other HFDS features still contribute their adverse influence in building the mapping vectors.

6.5. How does the parameter η affect the performance of proposed method?

The parameter η controls the influence of HFDS features in building the mapping vectors. In order to locate the ideal value range for η, we vary the parameter η from 0 to 1.0, run the proposed method W-SCL on six transfer tasks, and finally draw two curves of the averaged performance. 0 indicates that we donot use any HFDS features in Z_{HFDS} when building the mapping vectors, while 1.0 indicates that we donot limit the role of HFDS features.

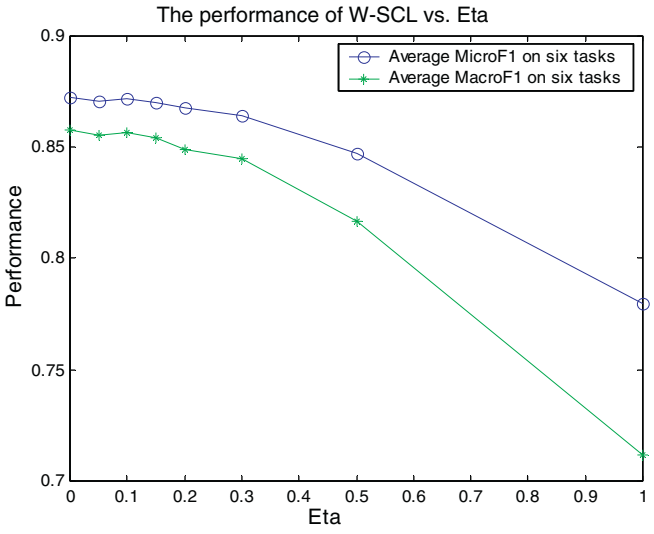

Fig. 5. The performance curves of W-SCL vs. η.

Note that for proposed W-SCL, the Z_{HFDS} is set to 200, and γ is set to 0.9. The performance curves of W-SCL vs. η is displayed in Fig. 5.

As expected, with the increase of η, the performance of proposed method descends fast. This result is in accordance with our analysis in Section 4: very few HFDS features occupy a large amount of *weight* in classification model while hardly carry corresponding sentiment information, and so they weaken the mapping ability of SCL model.

Secondly, it can be seen that proposed method yields excellent performance at the range [0,0.1]. This result indicates that small weight can decrease the adverse influence of HFDS features, and in the same time reserve certain useful sentiment information for classification.

6.6. How does the parameter γ affect the accuracy of proposed method?

The parameter γ controls the relative weight between the instances with the same labels as the pivot pvt_k and the other instances with the different labels as the pivot pvt_k. In this section we investigate whether increasing the weight of instances with the same label as the involved pivot pvt_k can boost the performance of SCL model. We vary the parameter γ from 0.1 to 1.0, run the proposed method W-SCL on six transfer tasks, and finally draw two curves of the averaged performance. Note that for proposed W-SCL, the Z_{HFDS} is set to 200 and η is set to 0.1. From Fig. 6, we can make two conclusions.

The first is that increasing the weight of instances with the same label as the involved pivot pvt_k can boost the performance of SCL model. As can be observed from Fig. 6, when increasing γ from 0.1 to 0.9, the average MicroF1 of W-SCL rises from about 71% to 87%.

The second conclusion is that only using same-label-as-the-pivot instances will miss some useful information when building mapping vectors. This result verifies our analysis in Section 5: firstly we cannot guarantee that the labels of instances are absolutely correct; secondly even on negative (positive) instances, there are some certain positive (negative) opinions.

7. Conclusion remarks

In this paper, we proposed a weighted SCL model (W-SCL) for domain adaptation in the context of sentiment analysis. On six

domain-transfer tasks, W-SCL consistently produces much better performance than the supervised, semi-supervised and transfer-learning baselines. As a result, we can say that proposed W-SCL model offers a better choice for sentiment-analysis applications that require high-precision classification but hardly have any labeled training data. The main contributions of this paper are three-folds:

First, we investigate the adverse influence of HFDS features on the mapping vectors using both empirical study and formally analysis. The investigation indicates that very few HFDS features occupy a relative large portion of *weight* in classification model while hardly carry corresponding sentiment information.

Secondly, we proposed Frequently Exclusively-occurring Entropy (FEE) to pick out HFDS features, and proposed feature-weighted SCL model (FW-SCL), which can adjust the influence of HFDS features in building correspondence. The experimental results indicate that decreasing the weight of HFDS features can boost the performance of traditional SCL with a wide margin.

Thirdly, in order to incorporate the labels of instances and pivots into the building of correspondence vectors, we proposed instance-weighted SCL model (IW-SCL). In the experiments, we found that instance-weighting strategy can further boost the performance of FW-SCL model.

Although proposed method indeed improves the classification accuracy, there is a lot of room for improvement. For example, FEE is not the best strategy for picking out HFDS features; whether other classifiers can work under this scheme; whether other methods other than RSR can be used to label some high-confidence instances. All these questions are waiting for our future efforts.

Acknowledgments

This work was mainly supported by two funds, i.e., 60933005 and 60803085.

References

Ando, R., & Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *JMLR, 6*, 1817–1853.
Aue, A., & Gamon, M. (2005). Customizing sentiment classifiers to new domains: A case study. In Proceedings of RANLP.
Blitzer, J., Dredze, M., & Pereir, F. (2007). Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. *ACL*.
Blitzer, J., McDonald, R., & Pereira, F. (2006). Domain adaptation with structural correspondence learning. *EMNLP*.
Chaovalit, P., & Zhou, L. (2005). Movie review mining: A comparison between supervised and unsupervised classification approaches. The 38th Hawaii International Conference on System Sciences.
Chelba, C., & Acero, A. (2004). Adaptation of maximum entropy capitalizer: Little data can help a lot. *EMNLP*.
Dai, W., Xue, G., Yang, Q., & Yu, Y. (2007). Transferring Naive Bayes classifiers for text classification. *AAAI*.
Daume, H., III, & Marcu, D. (2006). Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research, 26*, 101–126.
Du, W., Tan, S., Cheng, X. & Yun, X. (2010). Adapting information bottleneck method for automatic construction of domain-oriented sentiment lexicon. In Proceedings of WSDM.
Joachims, T. (1999). Transductive inference for text classification using support vector machines. *ICML*.
Lanquillon, C. (2000). Learning from labeled and unlabeled documents: A comparative study on semi-supervised text classification. *PKDD*.
McCallum, A., & Nigam, K. (1998). A comparison of event models for naive bayes text classification. In AAAI/ICML workshop on learning for text categorization.
Mullen, T., & Collier, N. (2004). Sentiment analysis using support vector machines with diverse information sources. *EMNLP*.
Nigam, K., McCallum, A., Thrun, S., & Mitchell, T. (1998). Learning to classify text from labeled and unlabeled documents. *AAAI*.
Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. *EMNLP*.
Raina, R., Ng, A., & Koller, D. (2005). Transfer learning for text classification. *NIPS*.
Tan, S., Cheng, X., Ghanem, M., Wang, B. & Xu, H. (2005). A Novel Refinement Approach for Text Categorization. In proceedings of CIKM.
Tan, S., Wu, G., Tang, H. & Cheng, X. (2007). A novel scheme for domain-transfer problem in the context of sentiment analysis. In proceedings of CIKM.
Tan, S., Cheng, X., Wang, Y. & Xu, H. (2009). Adapting Naive Bayes to domain adaptation for sentiment analysis. In proceedings of ECIR.
Tang, H., Tan, S., & Cheng, X. (2009). A survey on sentiment detection of reviews. *Expert Systems with Applications, 36*(7), 10760–10773.
Tan, S., & Zhang, J. (2008). An empirical study of sentiment analysis for Chinese documents. *Expert Systems with Applications, 34*(4), 2622–2629.